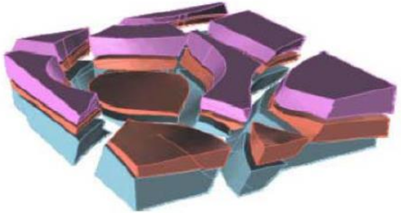
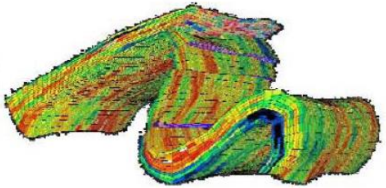
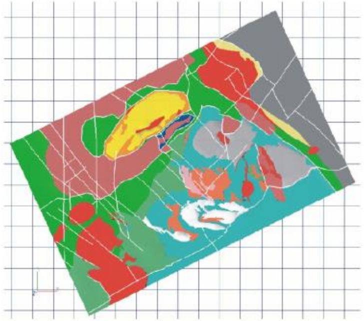
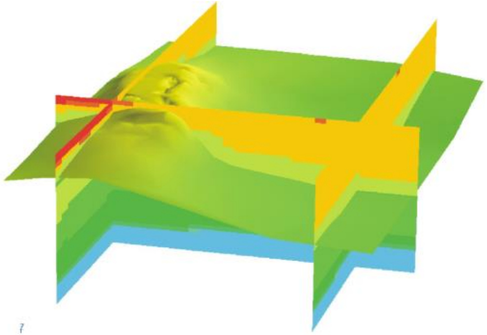
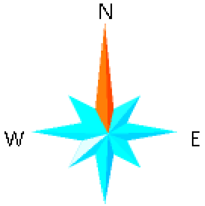
TECHNISCHE UNIVERSITÄT
BERGAKADEMIE FREIBERG
Die Ressourcenuniversität. Seit 1765.

3D Geomodellierung
2. Modellierung mit Skua-Gocad

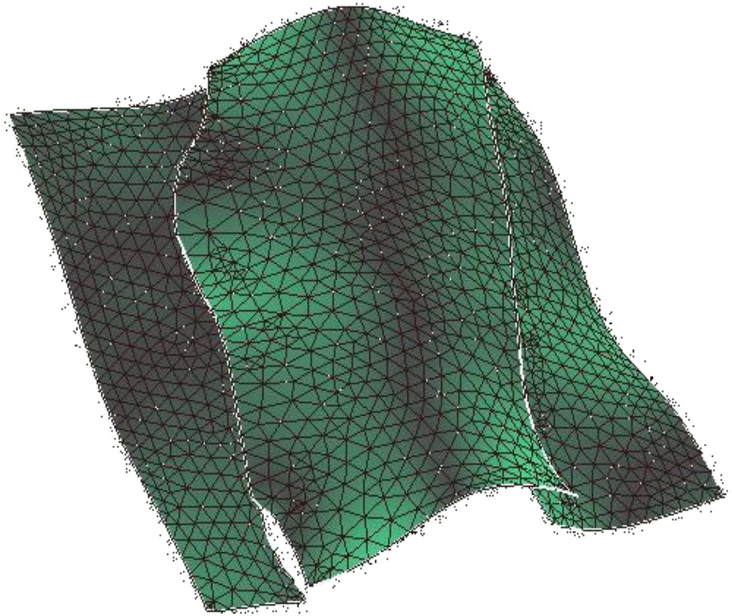





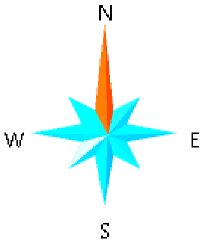



Modellierung mit Skua-Gocad

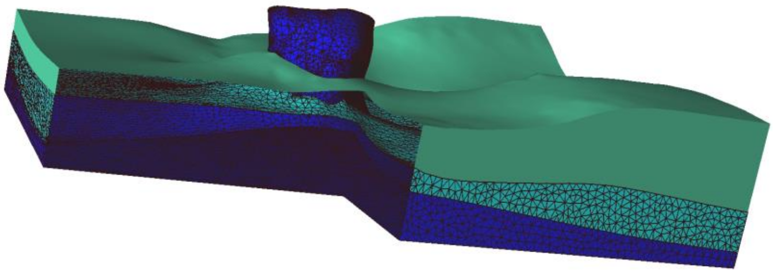
- Viele verschiedene Eingabedaten werden unterstützt.
- Zustandsmodellierung, basierend auf zellulärer Zerlegung natürlicher Objekte

- Grundlegende Objekttypen:
  - Punktwolken (PointSet)
  - Linienzüge (PolyLine)
  - Triangulierte Flächen (TSurf)



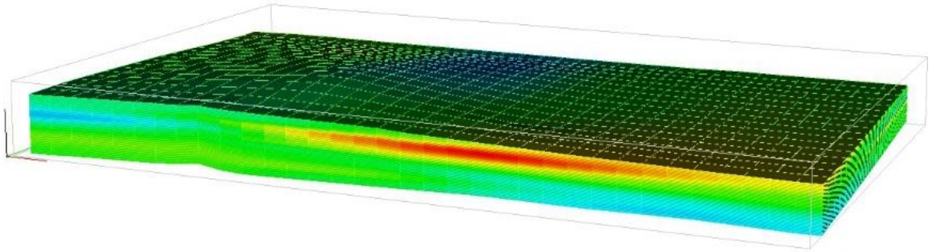


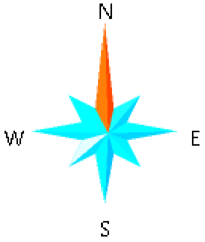
Modellierung mit Skua-Gocad



- Viele verschiedene Eingabedaten werden unterstützt.
- Zustandsmodellierung, basierend auf zellulärer Zerlegung natürlicher Objekte

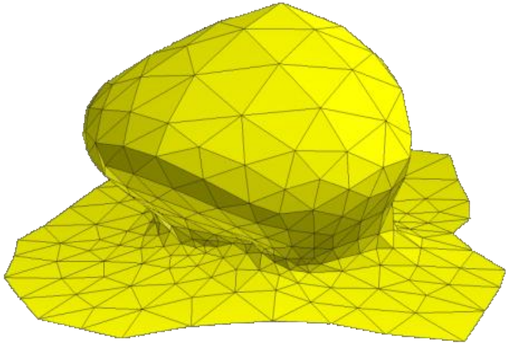
- Grundlegende Objekttypen:
  - Punktwolken (PointSet)
  - Linienzüge (PolyLine)
  - Triangulierte Flächen (TSurf)
  - Volumenkörper (Model3D)
  - Gitter (Voxet, SGrid)





Diskretes Datenmodell

Objekte werden an diskreten Punkten (Knoten) beschrieben und gespeichert



Topologie: Nachbarschaftsbeziehungen

Geometrie:

- Knotenpositionen, Distanzen, Winkel, Flächen, Volumina
- Invariant unter Translation, Rotationen oder Spiegelung

Constraints: Randbedingungen für Interpolation (geol. Daten oder Wissen)

Topologischer Graph $\mathcal{G}(\Omega, N)$

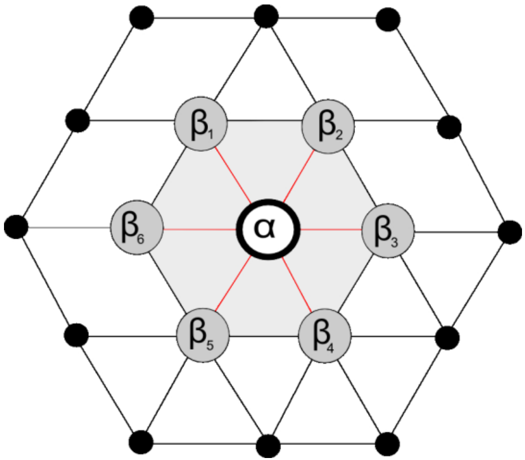
Der Ausgangspunkt für die Beschreibung eines diskreten Modells beruht auf der Annahme, das die Topologie eines Objektes über einen Graph $\mathcal{G}(\Omega, N)$ approximiert werden kann mit:

- Ω ist Menge der M Knoten des Graphen; $\Omega = \{1,2,\dots,\alpha,\dots,M\}$

- N bildet Ω auf eine Untermenge von Ω ab, mit

$$\{\beta \in N(\alpha)\} \Leftrightarrow \{\beta \text{ kann von } \alpha \text{ in } s(\alpha) \text{ Schritten erreicht werden}\}$$

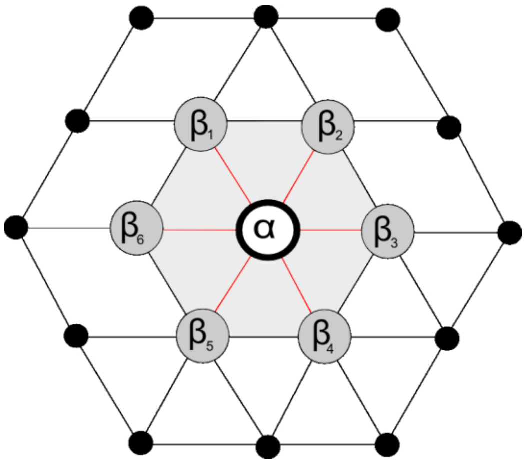
mit $s(\alpha) > 0$ (in der Praxis $s(\alpha) = 1$)



Topologischer Graph $\mathcal{G}(\boldsymbol{\Omega}, \boldsymbol{N})$

- $\mathcal{G}(\Omega, N)$ ist symmetrisch mit
$$\beta \in N(\alpha) \Leftrightarrow \alpha \in N(\beta)$$
… α und β sind „nah“ zueinander …

- $N(\alpha)$ wird als „Nachbarschaft von α“ bezeichnet und beinhaltet sowohl den Knoten α und die α umgebenden Knoten
$$N(\alpha) = \{\alpha, \beta_1, \beta_2, \dots\}$$



Diskretes topologisches Modell $\mathcal{M}^n(\boldsymbol{\Omega}, \boldsymbol{N}, \boldsymbol{\varphi}, \boldsymbol{\mathcal{C}})$

Geometrie und physikalische Eigenschaften sind im topol. Graphen $\mathcal{G}(\Omega, N)$ noch nicht berücksichtigt.

Diskretes topologisches Modell $\mathcal{M}^n(\boldsymbol{\Omega}, \boldsymbol{N}, \boldsymbol{\varphi}, \boldsymbol{\mathcal{C}})$

Geometrie und physikalische Eigenschaften sind im topol. Graphen $\mathcal{G}(\Omega, N)$ noch nicht berücksichtigt.

Diese Eigenschaften können als Serien von numerischen Funktionen, den Komponenten von φ, die auf den Knoten von $\mathcal{G}(\Omega, N)$ definiert sind, modelliert werden:

$$\varphi(\alpha) = \{\varphi^1(\alpha), \varphi^2(\alpha), \dots, \varphi^\nu(\alpha), \dots, \varphi^n(\alpha)\}\ \forall \alpha \in \Omega$$

Diskretes topologisches Modell $\mathcal{M}^n(\boldsymbol{\Omega}, \boldsymbol{N}, \boldsymbol{\varphi}, \boldsymbol{\mathcal{C}})$

In der Praxis beschreiben 3 Komponenten von $\varphi(\alpha)$, bezeichnet als $\{\varphi^x(\alpha), \varphi^y(\alpha), \varphi^z(\alpha)\}$, die Position des Knoten α im $\mathbb{R}^3$.

Die anderen Komponenten entsprechen den physikalischen Eigenschaften.

Diskretes topologisches Modell $\boldsymbol{\mathcal{M}^n(\Omega, N, \varphi, \mathcal{C})}$

Die Beschreibung eines topologisches Modell $\mathcal{M}^n(\Omega, N, \varphi, \mathcal{C})$ umfasst das Tripel aus

- dem Graph $\mathcal{G}(\Omega, N)$,
- den Funktionen φ, sowie
- einer Menge von „Constraints“ $\mathcal{C} = \{c_1, c_2, \dots\}$:

$$\mathcal{M}^n(\Omega, N, \varphi, \mathcal{C}) \equiv \{\mathcal{G}(\Omega, N), \varphi, \mathcal{C}\}$$

➢ Constraints sind letztendlich die Randbedingungen für eine Interpolation

Constraints $\mathcal{C}$

Die Menge der Constraints $\mathcal{C}$ lässt sich in drei Untermengen

$$\mathcal{C} = \mathcal{C}^{\cong} \cup \mathcal{C}^{=} \cup \mathcal{C}^{>}$$

Jeder Constraint $c \in \mathcal{C}$ wird als linear angenommen und entspricht einer von drei Formalisierungen mit den Koeffizienten $\{A_c^{\nu}(\alpha)\}$ und b_c:

Constraints $\mathcal{C}$

$\mathcal{C}^{\cong}$ „soft“ Constraint, Gleichheit muss im Sinne von „least squares“ berücksichtig werden (*control points*):

$$\{c \in \mathcal{C}^{\cong} \text{ erfüllt}\} \Leftrightarrow \sum_{\alpha \in \Omega} \sum_{\nu} A_c^{\nu}(\alpha) \cdot \varphi^{\nu}(\alpha) \cong b_c$$

$\mathcal{C}^{=}$ „hard“ Constraint, Gleichheit muss exakt berücksichtig werden (*control nodes*):

$$\{c \in \mathcal{C}^{=} \text{ erfüllt}\} \Leftrightarrow \sum_{\alpha \in \Omega} \sum_{\nu} A_c^{\nu}(\alpha) \cdot \varphi^{\nu}(\alpha) = b_c$$

$\mathcal{C}^{>}$ „hard“ Constraint, Ungleicheit muss exakt berücksichtigt werden

$$\{c \in \mathcal{C}^{>} \text{ erfüllt}\} \Leftrightarrow \sum_{\alpha \in \Omega} \sum_{\nu} A_c^{\nu}(\alpha) \cdot \varphi^{\nu}(\alpha) > b_c$$

Simple Constraints

Seien die Koeffizienten $\{A_c^\nu(\alpha)\}$ wie folgt definiert:

$$\exists \nu_c : A_c^\nu(\alpha) = 0 \ \forall \begin{cases} \nu \neq \nu_c \\ \& \\ \alpha \in \Omega \end{cases}$$

Dann ist c ein so genannter „simple“ Constraint.

$\Longrightarrow \{c \in \mathcal{C}^{\bowtie} \text{ erfüllt}\} \Leftrightarrow \sum_{\alpha \in \Omega} \sum_\nu A_c^\nu(\alpha) \cdot \varphi^\nu(\alpha) \bowtie b_c$ bezieht sich nur auf eine Komponente φ^{ν_c} von φ.

Beispiel für Constraints

In der Praxis, werden die Koeffizienten $\{A_c^\nu(\alpha)\}$ und b_c als Funktionen der verfügbaren Daten bestimmt.

Sei $\varphi^\eta(\alpha_0)$ gleich einem gegebenen Wert ϕ_0^η am Knoten α_0 kann diese Information wie folgt in einen Constraint $c \in \mathcal{C}^{\cong}$ oder $c \in \mathcal{C}^{=}$ umgesetzt werden:

$$\left| \begin{array}{l} A_c^\nu(\alpha) = \begin{cases} 1 \text{ wenn } \alpha = \alpha_0 \,\&\, \nu = \eta \\ \quad\quad 0 \text{ sonst} \end{cases} \\ \quad\quad\quad b_c = \phi_0^\eta \end{array} \right.$$

Diskretes topologisches Modell $\boldsymbol{\mathcal{M}^n(\Omega, N, \varphi, \mathcal{C})}$

Die Topologie $\mathcal{G}(\Omega, N)$ liegt in Skua-Gocad **unabhängig** von den Knoteneigenschaften $\varphi(\Omega)$ und den Constraints $\mathcal{C}$ vor.

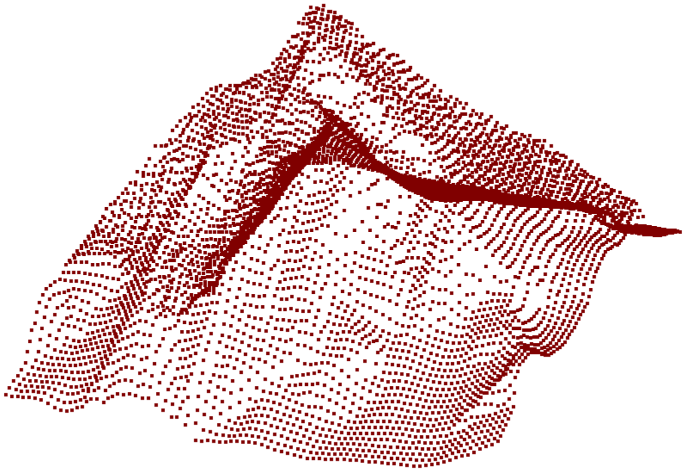
- In Skua-Gocad wird meistens **zuerst** die Topologie (z.B. Dreicksvermaschung einer Fläche) erstellt, **dann** werden die Constraints $\mathcal{C}$ definiert und **danach** werden Geometrie (endgültige Lage der Knoten/Punkte im Raum) und physikalische Parameter basierend auf den Constraints $\mathcal{C}$ interpoliert.

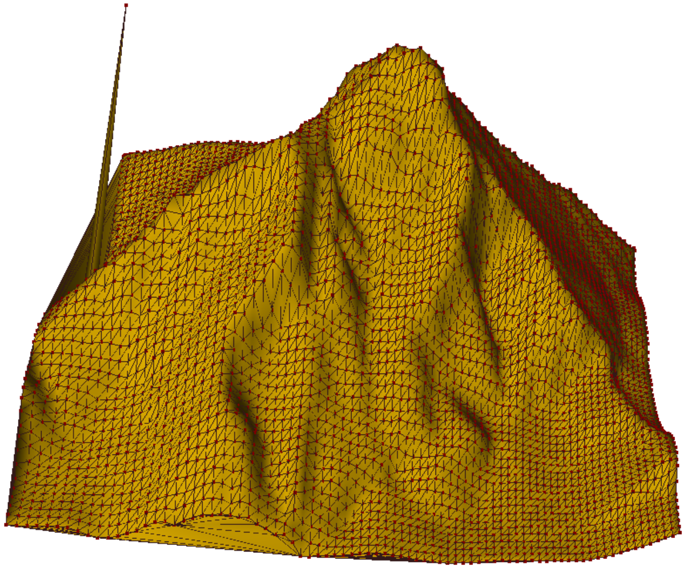
Direkte Triangulation

- Triangulation der Datenpunkte
- Dreiecksknoten stimmen mit Punktdaten überein
- Keine Constraints

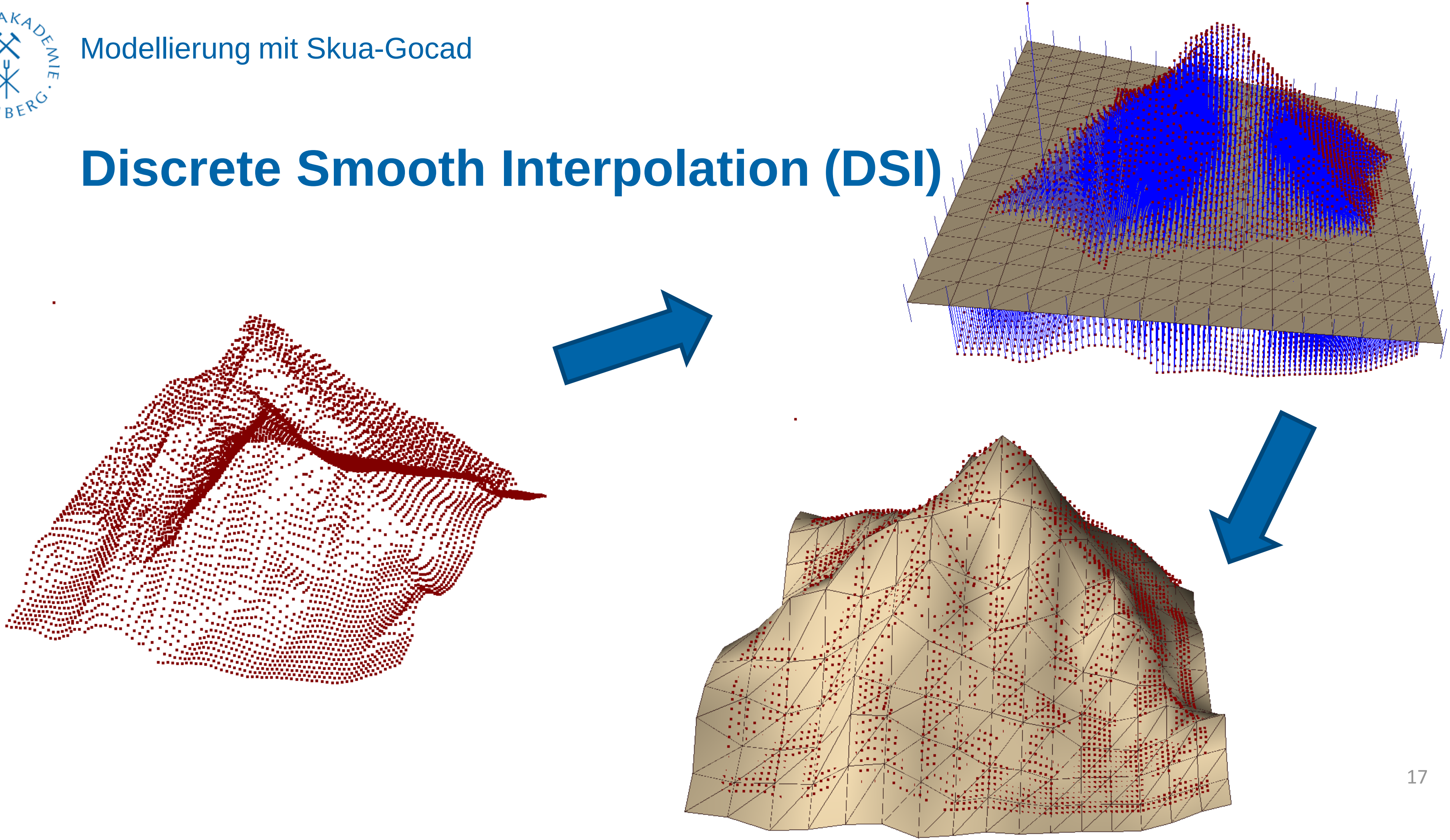
➢ Anzahl der Knoten entspricht der Anzahl der Datenpunkte
➢ Ausreißer in den Daten werden abgebildet

➢ … dient häufig als Grundlage für eine Modellierung über DSI …

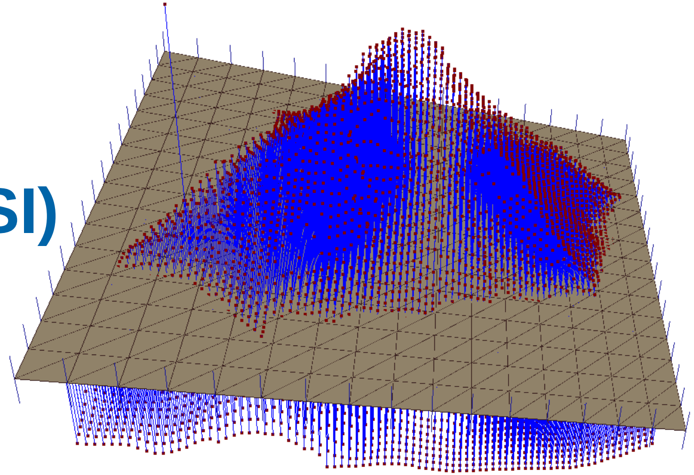




Discrete Smooth Interpolation (DSI)



Discrete Smooth Interpolation (DSI)



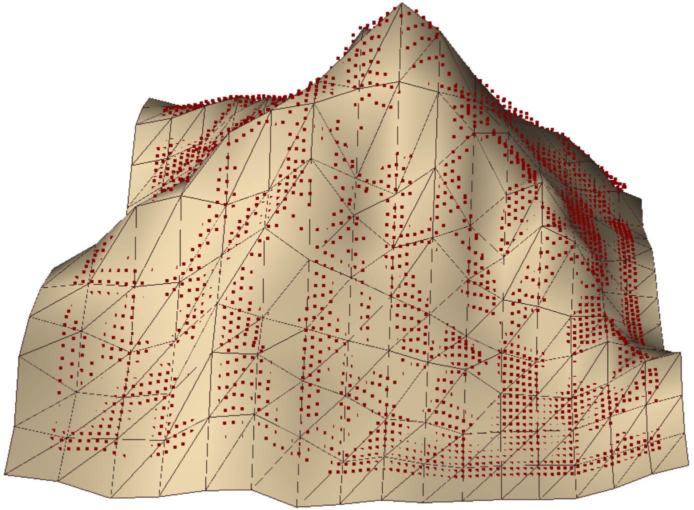
- Dreiecksknoten unabhängig von den Datenpunkten

- Datenpunkte als Constraints

- Trade-Off:
  Minimierung der globalen „Rauhigkeit“ vs Berücksichtigung der Constraints

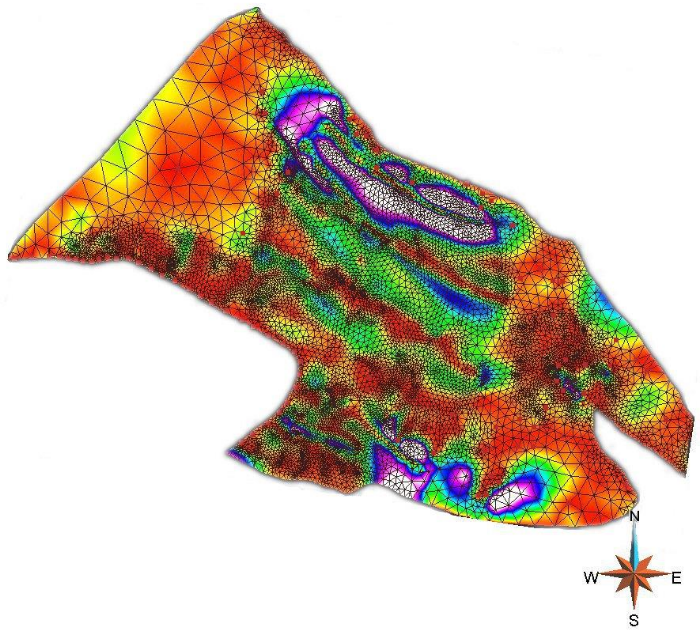
- „Günstige“ initiale Lage/Parameterwerte der Knoten ist vorteilhaft

➢ *medium plane* der Constraints



Take-home questions:

1) Welche drei Modellkomponenten sind für eine Modellierung in Skua-Gocad notwendig?
2) Worin unterscheiden sich Control Points (soft constraints) von Control Nodes (hard constraints)?
3) Warum lässt sich eine Fläche besser mit DSI als mit direkter Triangulation an gegebene Punktdaten anpassen?

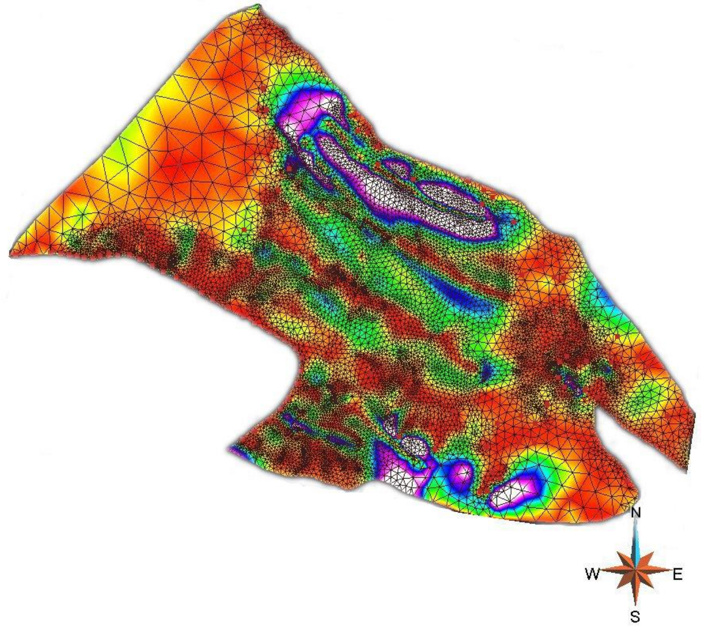


Institut für Geophysik und Geoinformatik
Dr. Peter Menzel
Gustav-Zeuner-Str. 12
09599 Freiberg

Tel. +49(0)3731 39-3815

Literatur:

Mallet, Jean-Laurent. 2002. *Geomodeling*. Oxford University Press.



Institut für Geophysik und Geoinformatik
Dr. Peter Menzel
Gustav-Zeuner-Str. 12
09599 Freiberg
Tel. +49(0)3731 39-3815